CST1204: Introduction to Databases

Jun Shan
Adjunct Professor
City Tech, CUNY

Week 3 Session 1

9/9/2019

Review

- Previous session review
  - Entity - Relationship
  - Functional Dependency
  - Normalization (first, second and third normal form)
  - Diagrams for database design

Learning Pattern in Next 2 Weeks

- Will give a scenario and ask you to
  - Identify entities and relationships and their primary key
  - Draw ERD
  - Normalize a bad design into 3NF
  - Write CREATE and DROP statements
  - Write further simple SQL statements

Agenda

- Introduction to SQL
- Introduction to Oracle
- Creating and dropping a table

Structured Query Language (SQL)

- ***THE*** standard way of relational database operation
  - Create/Maintain/Drop tables
  - Insert/Update/Delete/Query table data
- Descriptive, not procedural
- SQL has an ANSI Standard: Same across different vendors
  - Latest is SQL-2016

Introduction to Oracle

- CUNY Oracle SQL Workshop:
  - Direct query vs using a file
- Oracle Live SQL
- DESCRIBE statement
- Semicolon (;)

CREATE TABLE Statement

- CREATE statement (Ch 3 Pg 65)

```
CREATE TABLE REP (
REP_NUM CHAR(2) PRIMARY KEY,
LAST_NAME CHAR(15),
FIRST_NAME CHAR(15),
STREET CHAR(15),
CITY CHAR(15),
STATE CHAR(2),
POSTAL_CODE CHAR(5),
COMMISSION DECIMAL(7,2),
RATE DECIMAL(3,2) );
```

CREATE Statement Syntax

```
CREATE TABLE <table_name> (
<column_1_definition>
[,<column_2_definition>]
...
[,<column_n_definition>]
[,<table_properties>]
);

column_definition =
<column_name> <data_type> [<column_property>]
```

Naming Standard

- Same restrictions on table and column names
  - <= 30 characters
  - Start with a letter
  - Can contain letter, number, and underscore (_)
  - No space
- Example
  - Good: Student_ID, Teacher_Name, StudentID, N1104_Reservation, Name____
  - Bad: Student ID, Teacher_Salary_$, 2020_Sales_Goal, This_Column_Name_Is_Too_Long_In_Oracle

Data Types

- Defines what value the column can hold
- Common Data Types (Ch 3 Pg 71)
  - INT
  - FLOAT vs DECIMAL(p, s)
  - CHAR(n) vs VARCHAR(n)
    - Difference in storage and performance
  - DATE
    - Looks like a string but not a string

Convert Shorthand Rep to CREATE

```
CREATE TABLE <table_name> (
<column_1_definition>...
);
```

- Use entity/relationship name as `<table_name>`
- Use attribute name as column name, one attribute is one column
- Add column type to each column
- Add Primary Key constraint to primary key column(s)
- Add comma to the end of all columns except for the last column

DROP TABLE Statement

- DROP statement (Ch 3 Pg 69)

```
DROP TABLE REP;
```

- Statement syntax

```
DROP TABLE <table_name>
```

Hands-on

- Review previous homework, Ch2 Review Question 11
  a. Identify entities and relationships and their primary key
  b. Draw ERD
  c. Normalize a bad design into 3NF: Ch2 Review Question 15
  d. Write CREATE and DROP statements

Hands-on

1. Create all tables for TAL Distributors
    a. Identify entities and relationships and their primary key
    b. Draw ERD
    c. Normalize a bad design into 3NF: Ch2 Review Question 15
    d. Write CREATE and DROP statements
2. Save the script to your USB storage and drop the table
3. Load script to recreate the table
4. Error handling (Ch 3 Pg 68)

Homework

- Complete previous homework if not done yet: Check Blackboard.
- Write CREATE statements for all tables of TAL Distributors as shown in Figure 1-2 (all five tables) and draw ERD.
- Chapter 2 exercise questions for TAL Distributors.